JPL

Adaptive Network Communication

Shared Net IP Communications Layer

James McKelvey

Section 369

November 4, 2002

Introduction

JPL

- Developed as part of SharedNet 6
  - Distributed battlefield information system
  - Delivers "Common Tactical Picture" to distributed system of clients and servers
  - Runs over existing networks (EPLRS, SINCGARS, WARNET)
  - Provides a range of quality-of-service
- Network Characteristics
  - Bandwidth often limited
  - Heterogeneous and often unreliable
  - Nodes go in and out of communication, must be updated
- Requirements Fight One Another

Approach: Communication Abstractions

JPL

- Subsystem of SN
  - Independent, separately fieldable
- Two logical pieces:
  - Abstract logical "Channels" that specify delivery requirements and are protocol and network independent
  - Communication engine that picks the protocols and guarantees the level-of-service.
- Users interface primarily through the channel abstraction
- Written in standard Java

Capabilities: Channels

JPL

- Provide an abstract delivery mechanism called Channels.
  - Channels define the properties of message delivery including level-of-service, priority, and ordering.
  - Comm nodes are specified as Senders and/or Receivers of a channel.
    - Nodes may be added dynamically.
    - Channels support M receivers and N senders.
  - Channels do not specify delivery protocols.
  - User Interface is simplified, because channels define most of the communication properties.
- Channels are the primary user interface

Channel: Properties

JPL

- Priority
  - Higher-priority messages are processed ahead of lower
  - Higher priority messages get first crack at available bandwidth
- Ordering
  - Messages may optionally be held to guarantee delivery in the order they were originally sent by the originator
- Timeliness
  - Channels define a time interval during which the message is considered to be still valid. This may be infinite. The time interval is used to optimize message delivery.

Channel: Quality of Service

JPL

- Sender has increasing awareness of message delivery as service level increases
  - Datagram
    - No knowledge; message delivered if possible during specified timeout interval
  - Managed
    - No knowledge, except that if timeout is infinite, message will eventually be delivered if the sender does not restart
  - Acknowledged
    - Acknowledgement (positive or negative) delivered to sender; if timeout is infinite, acknowledgement will never be negative, and message will be delivered if the sender does not restart
  - Persistent
    - Similar to Acknowledged, but message delivery survives restart

Communication Engine

JPL

- Picks protocol for delivery
  - UDP, multicast, TCP
- Implements level-of-service under protocol
  - Rerequest of lost messages (whole or in part)
  - Even multicast may be reliable
- Implements message disassembly/assembly
  - Large messages split into fixed-size packets
  - Adapts packet size to network requirements
- Monitors and Controls Network Bandwidth
  - Hierarchical, I.e., controllable by subnet

Communication Engine (Cont.)

- Detects nodes no longer in communication
  - Messages held for delivery (until timed out)
- Keeps track of statistics
  - Message and bandwidth statistics
  - System and node state
  - Available to local and remote nodes
- Dynamically reconfigurable
  - Rule sets and parameters control decision making
  - Both may be dynamically determined or modified by administrators

Concept of Operation: Engine

JPL

- Distributed application
  - All nodes are peers
- Logic is decentralized
  - Decisions are localized
  - Decisions are driven by network state and configuration
- Client process creates comm object
  - Provides initial property list
  - Further changes can be made dynamically
- Client sets up communication
  - Elects to receive data on certain channels
  - Elects to send data on certain channels
  - Prepares to receive status messages

Protocol Selection

JPL

- Prefer multicast over point-to-point
  - When there are two or more known recipients
  - When number of potential recipients is unknown
- Prefer UDP over multicast
  - When there is one known recipient
- Prefer TCP
  - Only when channel is marked as high-data-rate
- Ultimately controlled by configuration
  - Any and all protocols may be disallowed, allowed, or forced
  - Preferences are controllable by subnet

Protocol Selection (cont.)

JPL

- Multiple protocols may be required
  - When subnets have different requirements
- Multiple transmissions may be required
  - When multiple protocols are used
  - When multicast cannot be used
  - When subnets specify different multicast addresses
- Transmissions are minimized
  - When multicast addresses span sets of subnets
  - Whenever possible

Reliable Communication Over UDP and Multicast

JPL

- Receiver is responsible
  - Sender may not know all recipients
  - Receivers request message retransmission
- Rerequests are multicast
  - When possible
  - All recipients monitor rerequests from other nodes
  - Rerequests are randomized and accumulated
  - The idea is to eliminate "rerequest storms" when multiple nodes miss the same packet
- Rerequested messages are sent to all
  - All original recipients receive the resent packets
  - Required because some nodes may have suppressed their rerequests

Summary

JPL

- Connection-less reliable communication
  - Provides detailed control
  - TCP supported for networks that can really use it
- Reliable multicast
  - Reduces network loading
- Configurable at subnet level
  - Supports heterogeneous networks
- Channels support most application topologies
  - Peer-to-peer
  - Client-server
  - Chat
- Channels define data requirements
  - Enhances network independence